UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/607,397	06/30/2000	Damon Barry	13768.132	9886

22913 7590 11/04/2003

WORKMAN NYDEGGER (F/K/A WORKMAN NYDEGGER & SEELEY)
60 EAST SOUTH TEMPLE
1000 EAGLE GATE TOWER
SALT LAKE CITY, UT 84111

EXAMINER	
KISS, ERIC B	

ART UNIT	PAPER NUMBER
2122	7

DATE MAILED: 11/04/2003

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/607,397	BARRY ET AL.	
	Examiner	Art Unit	
	Eric B. Kiss	2122	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *02 July 2003* .

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-4 and 7-27* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-4 and 7-27* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☒ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *30 June 2000* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

11)☐ The proposed drawing correction filed on _____ is: a)☐ approved b)☐ disapproved by the Examiner.

If approved, corrected drawings are required in reply to this Office action.

12)☐ The oath or declaration is objected to by the Examiner.

Priority under 35 U.S.C. §§ 119 and 120

13)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____ .

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

14)☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. § 119(e) (to a provisional application).

a) ☐ The translation of the foreign language provisional application has been received.

15)☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. §§ 120 and/or 121.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449) Paper No(s) _____ .
4) ☐ Interview Summary (PTO-413) Paper No(s). _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other:

DETAILED ACTION

1. The amendment of July 2, 2003, has been received and entered. Claims 1-4, 7-27 are pending.

Response to Amendment

2. The amendment to the specification appropriately addresses the objection to the drawings, as detailed in the previous office action. Accordingly, this objection is withdrawn in view of Applicant's amendment.

3. Applicant's amendment to claim 14 appropriately addresses the objection to claim 14, as detailed in the previous office action. Accordingly, this objection is withdrawn in view of Applicant's amendment.

Response to Arguments

4. Applicant's arguments filed July 2, 2003, have been fully considered but they are not persuasive.

a) It is noted, to clarify the record, that Applicant's statement in the first sentence of paragraph 2 on page 11 ("As pointed out...") reflects only the position taken by Applicant's representatives during the Interview of record, and does not reflect any agreements reached

during the course of the Interview or the position of the Office or the Examiner of record, with regard to the merits of the claimed invention.

b) In response to Applicant's arguments on pages 11-12, the collection of APIs included with the TETware product provide the capability to build/link test case files into a format executable by the test case controller, regardless of the source language, e.g., C, C++, Shell, Korn Shell, or Perl, used to code the test cases (see, for example, section 2.4 of TET_UG describing the API components as linkable object code). The TETware product further has the capability to handle test cases that were not developed to conform to one of the included APIs (see, for example, section 2.4.4 of TET_PG describing the handling of non-API test cases). Regardless of which language is used to program a test case, the test case has to be built/linked by the test case controller prior to execution (see, for example, the description of build mode in section 6.2.3 of TET_UG). When a test case is built, it is linked with the appropriate test case manager module and API, if the test case is API-compliant, and transformed into an executable test case having an interface with which the test case controller, in execution mode, can interact. Note that the compiled test case no longer has any dependency on the source language used to create the source code for that test case (the test case controller, if directed, proceeds to execute the specified compiled test cases). Therefore, the Examiner asserts that the collection of APIs included with the TETware product, along with a build mode invocation of the test case controller, create an interface to a test case that is language and format independent.

Claim Objections

5. Claim1 is objected to because of the following informalities: "insure" in line 15 should read --ensure--. Appropriate correction is required.

6. Claim 24 is objected to because of the following informalities: the "and" in line 15 appears to be unnecessary. Appropriate correction is required.

Claim Rejections - 35 USC § 102

7. The text of those sections of Title 35, U.S. Code not included in this action can be found in a prior Office action.

8. Claims 1, 2, 4, and 7-27 are rejected under 35 U.S.C. 102(b) as being anticipated by the TETware Release 3.3 software product (hereinafter TETware) released September 18, 1998 by The Open Group, as evidenced by: "TETware User Guide, Revision 1.2" (hereinafter TET_UG), "Release Notes for TETware Release 3.3" (hereinafter TET_RN), and "TETware Programmers Guide, Revision 1.2" (hereinafter TET_PG).

As per claim 1, TETware is disclosed with a computer system comprising:

a program module (test suite) **containing a plurality of individually accessible** test cases (see, for example, section 2.5.2 of TET_PG, which describes "Test scenario definitions"

that specify which test cases of a test suite are to be executed), each comprising a set of instructions for testing a feature of the computer program **through a language and format independent interface, at least some of the individually accessible test cases differing from one another in format** (the test cases are built and executed, regardless of their source language, through the same test case controller; see, for example, the description of build mode in section 6.2.3 of TET_UG; different test cases inherently differ from one another in format, e.g., at a low level, they comprise different bit patterns, and at a higher level, they comprise different instructions or parameters);

a harness, comprising a set of instructions that executes **a test case hierarchy** (test scenario) on the computer program **using the corresponding language and format independent interface of each individually accessible test case in the test case hierarchy** (test case controller; see sections 2.1 and 2.2 of TET_UG; the test cases are built and executed, regardless of their source language, through the same test case controller; see, for example, the description of build mode in section 6.2.3 of TET_UG);

a connector, comprising a set of instructions that **scans the plurality of test cases and extracts those test cases to be used to test the computer program to [ensure] that it processes as intended, the connector creating a hierarchy of test cases from those that are selected and extracted** (see, for example, section 2.5.2 of TET_PG, which describes "Test scenario definitions" that specify which test cases of a test suite are to be executed), and selectively integrates a generic interface between the one or more test cases and the harness regardless of the language or format in which the test cases were written (test case managers and

API libraries; see section 2.4 of TET_UG; see also section 2.4.4 of TET_PG describing the handling of non-API test cases); and

a processor for executing the one or more test cases, the harness and the connector (inherent in the operation of the UNIX and WINDOWS operating systems used to implement TETware; see section 1.1 of TET_UG).

As per claim 2, TETware is further disclosed with the set of instructions of the harness and the set of instructions of the connector utilizing an architecture that defines a means for accessing a resource over a network (see section 2.6.3 of TET_UG).

As per claim 4, TETware is disclosed with a method comprising:

the connector scanning the program module (test suite) **for one or more test cases of interest** (see, for example, section 2.5.2 of TET_PG, which describes "Test scenario definitions" that specify which test cases of a test suite are to be executed)**, each test case having a language and format independent interface for executing the test case on the computer program regardless of the language or format used to develop the test case** (the test cases are built and executed, regardless of their source language, through the same test case controller; see, for example, the description of build mode in section 6.2.3 of TET_UG);

the connector extracting the one or more test cases of interest from the program module (see, for example, section 2.5.2 of TET_PG, which describes "Test scenario definitions" that specify which test cases of a test suite are to be executed);

the connector organizing one or more test cases into a **test case** hierarchy (test suite structure; see section 2.2 of TET_UG; see, for example, section 2.5.2 of TET_PG, which describes "Test scenario definitions" that specify which test cases of a test suite are to be executed);

the connector interfacing a harness with the one or more test cases **of interest** (see section 6.4 of TET_UG; see, for example, section 2.5.2 of TET_PG, which describes "Test scenario definitions" that specify which test cases of a test suite are to be executed), wherein the interfacing allows the harness to recognize and execute the one or more test cases **of interest regardless of the language or format in which the one or more test cases of interest were developed** (test case controller; see sections 2.1 and 2.2 of TET_UG; the test cases are built and executed, regardless of their source language, through the same test case controller; see, for example, the description of build mode in section 6.2.3 of TET_UG); and

the harness traversing the test case hierarchy and executing **each of the one or more test cases of interest to the** computer program (see the description of the test case controller beginning on page 105 of TET_UG).

As per claim 7, TETware is further disclosed with a step of determining whether one or more of the test cases are identified as being deselected, wherein a deselected test case is not executed on the computer program (see, for example, the "-n" command line option of the test case controller on page 107 of TET_UG).

As per claim 8, TETware is further disclosed with one or more test cases comprising a test suite in the hierarchy (see section 2.2. of TET_UG).

As per claims 9, TETware is further disclosed with one or more test suites comprising a test module in the hierarchy (test scenario; see section 2.2 of TET_UG).

As per claims 10 and 11, TETware is further disclosed with excluding test cases determined to be deselected from a selection of a test suite or scenario (see, for example, the "-n" command line option of the test case controller on page 107 of TET_UG).

As per claims 12-14, TETware is further disclosed with the step of traversing further including executing the one or more test cases on a thread pool comprising one or more threads, and further discloses testing single-threaded and multi-threaded (thread-safe) models (see section 17.4 of TET_PG).

As per claims 15-17, these are computer-readable medium versions of the method discussed above (claim 4), wherein all limitations have been addressed as set forth above. Furthermore, the use of such a computer-readable medium containing executable code is inherently necessary for the operation of the UNIX and WINDOWS operating systems used to implement TETware (see section 1.1 of TET_UG).

As per claims 18 and 19, see the disclosure applied above in the rejection of claims 8 and 9, respectively.

As per claim 20, TETware is further disclosed with user-selected test cases (see the description of the test case controller and command line usage beginning on page 107 of TET_UG).

As per claims 21-23, see the disclosure applied above in the rejection of claims 12-14.

As per claim 24, TETware is disclosed with a method comprising:

identifying one or more **test cases** from a program module (see, for example, section 2.5.2 of TET_PG, which describes "Test scenario definitions" that specify which test cases of a test suite are to be executed), **each test case implementing a language and format independent interface for executing the test case on a computer program regardless of the language or format used to develop the test case** (the test cases are built and executed, regardless of their source language, through the same test case controller; see, for example, the description of build mode in section 6.2.3 of TET_UG);

translating the identified **one or more test cases** into a **test case** hierarchy (a test scenario see, for example, section 2.5.2 of TET_PG, which describes "Test scenario definitions" that specify which test cases of a test suite are to be executed);

interfacing the **test case** hierarchy **in order to recognize and execute the one or more test cases regardless of the language or format in which the one or more test cases were written** (test case controller; see sections 2.1, 2.2, and 2.4 of TET_UG; the test cases are built and executed, regardless of their source language, through the same test case controller; see, for example, the description of build mode in section 6.2.3 of TET_UG); and

executing **each of the one or more test cases in the test case hierarchy to test** the computer program regardless of the language or format in which the **one or more test cases** were written (test case managers and API libraries; see section 2.4 of TET_UG; see also section 2.4.4 of TET_PG describing the handling of non-API test cases; the test cases are built and executed, regardless of their source language, through the same test case controller; see, for example, the description of build mode in section 6.2.3 of TET_UG).

As per claims 25-27, see the disclosure applied above in the rejection of claims 12-14.

Claim Rejections - 35 USC § 103

9. The text of those sections of Title 35, U.S. Code not included in this action can be found in a prior Office action.

10. Claim 3 is rejected under 35 U.S.C. 103(a) as being unpatentable over TETware and the associated cited documentation as applied to claim 1 above, and further in view of U.S. Patent No. 6,505,342 to Hartmann et al.

As per claim 3, TETware is disclosed with such a system (see disclosure applied above to claim 1), but is not expressly disclosed with a COM technology architecture. However, Hartmann et al. teach a system for testing components that use middleware, such as COM/DCOM (see column 2, line 61 through column 3, line 4). Therefore, it would have been obvious to one having ordinary skill in the computer art at the time the invention was made to modify the system of TETware to include a COM architecture as per the teaching of Hartmann et al. One would be motivate to do so to gain the advantage of supporting and testing implementations in a standardized object-oriented middleware.

Conclusion

11. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure. A copy of the pages of the TET_PG reference that were not cited in the original office action is being provided herewith.

12. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

13. Any inquiry concerning this communication or earlier communications from the Examiner should be directed to Eric B. Kiss whose telephone number is (703) 305-7737. The Examiner can normally be reached on Tue. - Fri., 7:30 am - 5:00 pm. The Examiner can also be reached on alternate Mondays.

If attempts to reach the Examiner by telephone are unsuccessful, the Examiner's supervisor, Tuan Dam, can be reached on (703) 305-4552. The fax phone number for the organization where this application or proceeding is assigned is (703) 872-9306.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is (703) 305-3900.

EBK
October 21, 2003

ANTONY NGUYEN-BA
PRIMARY EXAMINER